

EXPERIMENTO 5 - EEPROM

Sistema Microcontrolados

Serial.available()

- ▶ Retorna o número de bytes (caracteres) disponíveis para leitura da porta serial. Esses são dados que já chegaram e foram guardados no buffer de recebimento (o qual armazena 64 bytes).
- ▶ Sintaxe
 - ▶ `Serial.available()`
- ▶ Retorna
 - ▶ O número de bytes disponíveis para leitura.

```
int incomingByte = 0; // para dados recebidos na porta serial

void setup()
{
    Serial.begin(9600); // abre a porta serial, taxa de transmissão 9600 bps
}

void loop() {
    // enviar resposta apenas quando receber dados:
    if (Serial.available() > 0) {
        // lê o dado recebido:
        incomingByte = Serial.read();
        // responder o que foi recebido:
        Serial.print("Eu recebi: ");
        Serial.println(incomingByte, DEC);
    }
}
```

EEPROM.write()

- ▶ Description
 - ▶ Write a byte to the EEPROM.
- ▶ Syntax
 - ▶ `EEPROM.write(address, value)`
- ▶ Parameters
 - ▶ address: the location to write to, starting from 0 (*int*)
 - ▶ value: the value to write, from 0 to 255 (*byte*)

EEPROM.read()

- ▶ Description
 - ▶ Reads a byte from the EEPROM. Locations that have never been written to have the value of 255.
- ▶ Syntax
 - ▶ `EEPROM.read(address)`
- ▶ Parameters
 - ▶ `address`: the location to read from, starting from 0 (*int*)

```
#include <EEPROM.h>
```

```
int addr = 0;
```

```
char ch;
```

```
void setup(){
```

```
    Serial.begin(9600);
```

```
    ch = EEPROM.read(addr);
```

```
}
```

```
void loop(){
```

```
    if (Serial.available() > 0){
```

```
        ch = Serial.read();
```

```
        EEPROM.write(0, ch);
```

```
        Serial.println(ch);
```

```
    }
```

```
    Serial.println(ch);
```

```
    delay(1000);
```

```
}
```